

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2005-135024

(43)Date of publication of application : 26.05.2005

(51)Int.Cl.

G06F 13/00

(21)Application number : 2003-367895 (71)Applicant : ANDO KAZUNORI

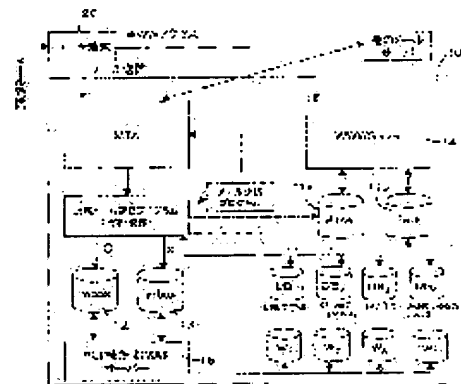
(22)Date of filing : 28.10.2003 (72)Inventor : ANDO KAZUNORI

(54) ANTI-SPAM METHOD AND ANTI-SPAM PROGRAM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide an anti-spam method and a spam determination program that can reliably detect spam according to a URL address included in the spam.

SOLUTION: The anti-spam method for email sending/receiving comprises a mail analysis step of cracking an email into parts in the smallest unit of mail structure according to separator character strings included in the mail body, a digital fingerprint acquisition step of acquiring a digital fingerprint from every cracked part, a URL information acquisition step of acquiring URL information included in the part, a reference step of referring to a database storing digital fingerprints or URL information acquired/stored from past emails, and a delivery stop step of determining that the email is spam and stopping the delivery of the email if the digital fingerprint acquired in the digital fingerprint acquisition step or the URL information acquired in the URL information acquisition step is stored in the database as a digital fingerprint or URL information related to spam.



(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2005-135024

(P2005-135024A)

(43) 公開日 平成17年5月26日(2005.5.26)

(51) Int.Cl.⁷
G06F 13/00

F I
G06F 13/00 610Q

テーマコード(参考)

審査請求 未請求 請求項の数 12 O L (全 22 頁)

(21) 出願番号 特願2003-367895 (P2003-367895)
(22) 出願日 平成15年10月28日(2003.10.28)

(特許庁注:以下のものは登録商標)

1. UNIX

(71) 出願人 503396088
安藤 一憲
東京都中央区新富1丁目6-10 メゾン
・ド・ヴィレ銀座東906
(74) 代理人 100088580
弁理士 秋山 敦
(74) 代理人 100111109
弁理士 城田 百合子
(72) 発明者 安藤 一憲
東京都中央区新富1丁目6-10 メゾン
・ド・ヴィレ銀座東906

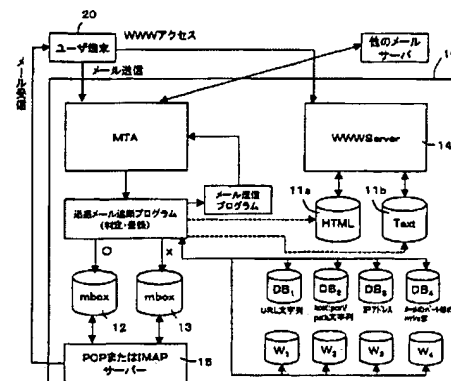
(54) 【発明の名称】 迷惑メール遮断方法及び迷惑メール遮断プログラム

(57) 【要約】

【課題】 本発明の目的は、迷惑メールに含まれるURLアドレスに基づいて、確実に迷惑メールを検出することを可能とした、迷惑メール遮断方法及び迷惑メール判定プログラムを提供する。

【解決手段】 電子メールの送受信における迷惑メール遮断方法であって、電子メールを、メール本文に含まれるセパレータ文字列に従って、メール構造の最小単位であるパートに分解するメール解析工程と、分解したパート毎に電子指紋を取得する電子指紋取得工程と、パートに含まれるURL情報を取得するURL情報取得工程と、過去の電子メールから取得・蓄積された電子指紋またはURL情報が格納されたデータベースを参照する参照工程と、電子指紋取得工程で取得された電子指紋またはURL情報取得工程で取得されたURL情報が、データベースに迷惑メールに関連する電子指紋またはURL情報として格納されていた場合に、電子メールを迷惑メールと判定して該電子メールの配信を停止する配信停止工程と、を備えている。

【選択図】 図2



【特許請求の範囲】**【請求項1】**

電子メールの送受信における迷惑メール遮断方法であって、

前記電子メールを、メール本文に含まれるセパレータ文字列に従って、メール構造の最小単位であるパートに分解するメール解析工程と、

前記分解したパート毎に電子指紋を取得する電子指紋取得工程と、

前記パートに含まれるURL情報を取得するURL情報取得工程と、

過去の電子メールから取得・蓄積された電子指紋またはURL情報が格納されたデータベースを参照するデータベース参照工程と、

前記電子指紋取得工程で取得された電子指紋または前記URL情報取得工程で取得されたURL情報が、前記データベースに迷惑メールに関連する電子指紋またはURL情報として格納されていた場合に、前記電子メールを迷惑メールと判定して該電子メールの配信を停止する配信停止工程と、を備えたことを特徴とする迷惑メール遮断方法。

【請求項2】

前記URL情報取得工程では、符号化されたURL文字列を復号する処理、復号されたURL文字列を分解する処理、前記分解された部分に基づいてホスト名、ポート番号、パス名からなる文字列を生成する処理、前記ホスト名に基づいてIPアドレスを取得する処理がなされることを特徴とする請求項1記載の迷惑メール遮断方法。

【請求項3】

電子メールの送受信における迷惑メール遮断方法であって、

前記電子メールを、メール本文に含まれるセパレータ文字列に従って、メール構造の最小単位であるパートに分解するメール解析工程と、

前記分解したパート毎に電子指紋を取得する電子指紋取得工程と、

前記パートに含まれるURL情報を取得するURL情報取得工程と、

過去の電子メールから取得・蓄積された電子指紋またはURL情報が格納されたデータベースを参照するデータベース参照工程と、

前記電子指紋取得工程で取得された電子指紋または前記URL情報取得工程で取得されたURL情報が既に存在した場合には、データを更新するデータ更新処理を行い、前記電子指紋取得工程で取得された電子指紋または前記URL情報取得工程で取得されたURL情報が存在しなかった場合には、新規データとして登録する新規データ登録処理を行うことを特徴とする迷惑メール遮断方法。

【請求項4】

前記データ更新処理または新規データ登録処理では、前記電子指紋またはURL情報の重複登録回数、最終更新日時、最初に登場した電子メールのアーカイブ名が登録されることを特徴とする請求項3記載の迷惑メール遮断方法。

【請求項5】

前記最終更新日時から現時点までの期間に応じて、登録の古い順からデータの削除を行うことを特徴とする請求項4記載の迷惑メール遮断方法。

【請求項6】

前記電子メール自体の情報と、前記メール構造に関する情報と、前記電子指紋に関する情報と、前記URL情報と、のうち少なくとも一つをHTMLファイルまたはテキストファイルとして出力する工程と、

該HTMLファイルまたはテキストファイルをウェブサーバを介して外部に表示する工程と、を備えたことを特徴とする請求項1または3記載の迷惑メール遮断方法。

【請求項7】

前記IPアドレスをDNSサーバに適用可能なブラックリストとして出力する処理がなされることを特徴とする請求項2記載の迷惑メール遮断方法。

【請求項8】

受信した電子メールのヘッダ情報を取得するステップと、

前記電子メールの本文のハッシュ値を計算するステップと、
前記電子メールの本文の復号を行うステップと、
前記電子メールがマルチパート構造の場合に、セパレータ文字列に従って各パートを切り出すステップと、
前記切り出された各パートについて、ヘッダ情報の取得、本文のハッシュ値の計算、本文の復号、各パートの切り出しを繰り返し行うステップと、
前記各パートが可読の場合にURL情報を抽出するステップと、
を備えたことを特徴とする迷惑メール遮断プログラム。

【請求項9】

受信した電子メールからURL文字列を取得するステップと、
前記URL文字列が符号化されている場合に復号化するステップと、
前記復号化されたURL文字列を、スキーム、ユーザ情報、ホスト名、ポート番号、パス名、クエリーに分解するステップと、
前記分解された部分に基づいて、ホスト名、ポート番号、パス名からなる文字列を生成するステップと、
前記ホスト名からIPアドレスを取得するステップと、
を備えたことを特徴とする迷惑メール遮断プログラム。

【請求項10】

前記ハッシュ値、前記復号化されたURL文字列、前記ホスト名、ポート番号、パス名からなる文字列、前記IPアドレスを迷惑メール判定のための判定情報とし、
新たに受信した電子メールの判定情報と、過去の電子メールから取得・蓄積された判定情報とを対比させるステップと、
前記新たに受信した電子メールの判定情報のうち少なくとも1つが、前記過去の電子メールから取得・蓄積された判定情報に合致した場合に、前記新たに受信した電子メールを迷惑メールとして判定するステップと、を備えたことを特徴とする請求項8または9記載の迷惑メール遮断プログラム。

【請求項11】

前記ハッシュ値、前記復号化されたURL文字列、前記ホスト名、ポート番号、パス名からなる文字列、前記IPアドレスを迷惑メール判定のための判定情報とし、
新たに受信した電子メールの判定情報と、過去の電子メールから取得・蓄積された判定情報のデータベースとを対比させるステップと、
新たに受信した電子メールの判定情報のうち少なくとも一つが、過去の電子メールから取得・蓄積された判定情報と合致した場合に、前記データベースのカウントを1繰り上げるとともに、更新日時を更新して再登録するステップと、
新たに受信した電子メールの判定情報が、過去の電子メールから取得・蓄積された判定情報と合致しない場合に、前記新たに受信した電子メールの判定情報を前記データベースに新規登録するステップと、
を備えたことを特徴とする請求項8または9記載の迷惑メール遮断プログラム。

【請求項12】

前記各ステップにおける処理の結果をHTMLファイルまたはテキストファイルに保存するステップと、
前記HTMLファイルまたはテキストファイルをウェブ上で参照するためのURLをユーザに送信するステップと、を備えたことを特徴とする請求項8乃至11いずれか記載の迷惑メール遮断プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、迷惑メール遮断方法に係り、特に、メールサーバにおいて迷惑メールのユーザへの配信を遮断する迷惑メール遮断方法と、この迷惑メール遮断方法において実行される迷惑メール遮断プログラムに関する。

【背景技術】

【0002】

近年、インターネット等の通信網を利用した電子メールの送受信において、受信者側の要・不要に係わらず、不特定多数の人に一方的に大量のメールが発信される事態が多発し、社会問題となっている。このようなメールは、受信者側に不快感を与えるとともに、不要なパケット料金の支払い等が発生するものであるため、迷惑メール又はスパムメールと呼ばれている。

【0003】

このような迷惑メールへの対策として、受信者側へ迷惑メールが配信される前に、迷惑メールを自動的に排除・処分するシステムが提案されている（例えば、特許文献1参照）。

【特許文献1】特開2003-131999号公報（第3-4頁）

【発明の開示】

【発明が解決しようとする課題】

【0004】

特許文献1に記載された技術は、携帯電話において、メールの本文中にURL（Uniform Resource Locator）が含まれるか否かを判定し、URLが含まれている電子メールを抽出する。ユーザは、ごみ箱メモリに保存された電子メールを参照し、迷惑メールでないものがあつた場合には、その電子メールをメール保存メモリに移動させる。このようにして、受信者側に配信された電子メールのなかから、迷惑メールを抽出することを可能としている。

しかし、上記従来技術によれば、送信側でURLに部分的に改変が加えられてしまった場合には、対応することができないという問題があつた。ドメイン名は容易に更新できるため、上記従来技術では迷惑メールの遮断を確実に行うことは不可能であつた。

【0005】

迷惑メールを遮断する方法として、上記特許文献1のような技術の他に、レイティング方式と呼ばれる技術がある。レイティング方式とは、インターネット上の各ホームページに対して、「アダルトサイト」、「暴力サイト」などのラベルを付けておき、フィルタリングソフトがそれらのラベルに基づいて、自動的にホームページへのアクセスを制限する方式である。

しかし、上記方法では、レイティングされていないホームページについては対応することができないという問題があつた。また、この方法においても、送信側でURLに部分的に改変が加えられてしまった場合には、対応することができないという問題があつた。

【0006】

さらにまた、他の迷惑メールの遮断方法として、メールの本文中に所定の文字を見つけた場合に、そのメールを迷惑メールと判定する方法が知られている。

現在では、一方的に送りつけられるメールに対して、その表題欄に「未承諾広告*」或いは「! 広告!」を付記することが義務づけられている。したがって、メール文中に、「未承諾広告*」或いは「! 広告!」の文字を発見した場合には、迷惑メールとして判定するものである。しかし、上記方法では、言語依存が激しいため、全ての言語で効果を上げるのは非常に難しい。

【0007】

本発明の目的は、迷惑メールに含まれるURLアドレスに基づいて、確実に迷惑メールを検出することを可能とした、迷惑メール遮断方法及び迷惑メール判定プログラムを提供することにある。

【課題を解決するための手段】

【0008】

前記課題は、請求項1に係る発明によれば、電子メールの送受信における迷惑メール遮断方法であつて、前記電子メールを、メール本文に含まれるセパレータ文字列に従つて、メール構造の最小単位であるパートに分解するメール解析工程と、前記分解したパート毎に電子指紋を取得する電子指紋取得工程と、前記パートに含まれるURL情報を取得する

URL情報取得工程と、過去の電子メールから取得・蓄積された電子指紋またはURL情報が格納されたデータベースを参照する参照工程と、前記電子指紋取得工程で取得された電子指紋または前記URL情報取得工程で取得されたURL情報が、前記データベースに迷惑メールに関連する電子指紋またはURL情報として格納されていた場合に、前記電子メールを迷惑メールと判定して該電子メールの配信を停止する配信停止工程と、を備えたことにより解決される。

【0009】

このように、本発明の迷惑メール遮断方法によれば、電子メールの各パートの電子指紋と、電子メールに含まれるURL情報に基づいて、ユーザ宛に繰り返し配送される迷惑メールを判定するようにされている。このため、入れ子構造のパートにURL情報が隠蔽されていたり、URL情報が符号化等により偽装されているときにも、ユーザ宛に繰り返し送付される情報を把握し、迷惑メールを確実に遮断することが可能となる。

【0010】

前記URL情報取得工程では、符号化されたURL文字列を復号する処理、復号されたURL文字列を分解する処理、前記分解された部分に基づいてホスト名、ポート番号、パス名からなる文字列を生成する処理、前記ホスト名に基づいてIPアドレスを取得する処理がなされる。

上記処理により、URL情報が符号化により偽装されている場合であっても、復号化することにより偽装を見破ることが可能となる。また、改変することが難しい「ホスト名、ポート番号、パス名からなる文字列」を生成したり、IPアドレスを取得することにより、繰り返し送付される情報を確実に発見することが可能となる。

【0011】

また、本発明におけるデータベースの構築は、請求項3に記載のように行う。

すなわち、電子メールの送受信における迷惑メール遮断方法であって、前記電子メールを、メール本文に含まれるセパレータ文字列に従って、メール構造の最小単位であるパートに分解するメール解析工程と、前記分解したパート毎に電子指紋を取得する電子指紋取得工程と、前記パートに含まれるURL情報を取得するURL情報取得工程と、過去の電子メールから取得・蓄積された電子指紋またはURL情報が格納されたデータベースを参照するデータベース参照工程と、前記電子指紋取得工程で取得された電子指紋または前記URL情報取得工程で取得されたURL情報が既に存在した場合には、データを更新するデータ更新処理を行い、前記電子指紋取得工程で取得された電子指紋または前記URL情報取得工程で取得されたURL情報が存在しなかった場合には、新規データとして登録する新規データ登録処理を行う。

【0012】

前記データ更新処理または新規データ登録処理では、前記電子指紋またはURL情報の重複登録回数、最終更新日時、最初に登場した電子メールのアーカイブ名が登録される。そして、前記最終更新日時から現時点までの期間に応じて、登録の古い順からデータの削除が行われる。

【0013】

また、前記電子メール自体の情報と、前記メール構造に関する情報と、前記電子指紋に関する情報と、前記URL情報のうち少なくとも一つをHTMLファイルまたはテキストファイルとして出力する工程と、該HTMLファイルまたはテキストファイルをウェブサーバを介して外部に表示する工程とを備えた構成とすると、外部からでも解析結果を参照することが可能となる。

【0014】

また、前記IPアドレスをDNSサーバに適用可能なブラックリストとして出力する処理がなされると、このブラックリストをDNS経由で参照することで、リモートホストでも迷惑メールの遮断が可能となる。

【0015】

本発明の迷惑メール遮断プログラムは、電子メールの解析と、URL情報の抽出を行う

ために、次の各ステップを備えている。

すなわち、受信した電子メールのヘッダ情報を取得するステップと、前記電子メールの本文のハッシュ値を計算するステップと、前記電子メールの本文の復号を行うステップと、前記電子メールがマルチパート構造の場合に、セパレータ文字列に従って各パートを切り出すステップと、前記切り出された各パートについて、ヘッダ情報の取得、本文のハッシュ値の計算、本文の復号、各パートの切り出しを繰り返し行うステップと、前記各パートが可読の場合にURL情報を抽出するステップと、を備えている。

【0016】

また、本発明の迷惑メール遮断プログラムは、URL情報の解析を行うために、次の各ステップを備えている。

すなわち、受信した電子メールからURL文字列を取得するステップと、前記URL文字列が符号化されている場合に復号化するステップと、前記復号化されたURL文字列を、スキーム、ユーザ情報、ホスト名、ポート番号、パス名、クエリーに分解するステップと、前記分解された部分に基づいて、ホスト名、ポート番号、パス名からなる文字列を生成するステップと、前記ホスト名からIPアドレスを取得するステップと、を備えている。

【0017】

さらに、本発明の迷惑メール遮断プログラムは、受信した電子メールを迷惑メールとして判定するために、次の各ステップを備えている。

すなわち、前記ハッシュ値、前記復号化されたURL文字列、前記ホスト名、ポート番号、パス名からなる文字列、前記IPアドレスを迷惑メール判定のための判定情報とし、新たに受信した電子メールの判定情報と、過去の電子メールから取得・蓄積された判定情報とを対比させるステップと、前記新たに受信した電子メールの判定情報のうち少なくとも1つが、前記過去の電子メールから取得・蓄積された判定情報に合致した場合に、前記新たに受信した電子メールを迷惑メールとして判定するステップを備えている。

【0018】

さらにまた、本発明の迷惑メール遮断プログラムは、データベースを構築するために、次の各ステップを備えている。

すなわち、前記ハッシュ値、前記復号化されたURL文字列、前記ホスト名、ポート番号、パス名からなる文字列、前記IPアドレスを迷惑メール判定のための判定情報とし、新たに受信した電子メールの判定情報と、過去の電子メールから取得・蓄積された判定情報のデータベースとを対比させるステップと、新たに受信した電子メールの判定情報のうち少なくとも一つが、過去の電子メールから取得・蓄積された判定情報と合致した場合に、前記データベースのカウントを1繰り上げるとともに、更新日時を更新して再登録するステップと、新たに受信した電子メールの判定情報が、過去の電子メールから取得・蓄積された判定情報と合致しない場合に、前記新たに受信した電子メールの判定情報を前記データベースに新規登録するステップと、を備えている。

【0019】

さらにまた、本発明の迷惑メール遮断プログラムは、解析結果をユーザへ報告するために、次の各ステップを備えている。

すなわち、前記各ステップにおける処理の結果をHTMLファイルまたはテキストファイルに保存するステップと、前記HTMLファイルまたはテキストファイルをウェブ上で参照するためのURLをユーザに送信するステップと、を備えている。

【発明の効果】

【0020】

本発明の迷惑メール遮断方法及び迷惑メール遮断プログラムによれば、マルチパート構造の電子メールの中にURL情報が隠蔽されている場合や、URL情報が偽装されている場合であっても、電子メールの解析及びURL情報の解析により、これらの隠蔽や偽装を見破り、迷惑メールを確実に検出することが可能となる。

【発明を実施するための最良の形態】

【0021】

以下、本発明の一実施の形態を図面に基づいて説明する。なお、以下に説明する部材、配置等は本発明を限定するものでなく、本発明の趣旨の範囲内で種々改変することができるものである。

【0022】

図1は一般的な電子メール送受信のシステム構成を示す説明図、図2は本発明の実施の形態における迷惑メール遮断方法及びプログラムが実行されるシステムの構成図、図3は迷惑メール検出の手順を示すブロック図、図4はメール受信からメールボックス格納までの流れを示す説明図、図5はメール受信から解析結果提示までの流れを示す説明図、図6はプログラムのオプションの例を示す一覧、図7は実際のaliasの例を示す一覧、図8乃至図10はMIMEメールの構造を示す説明図、図11はマルチパートの電子メールの復号化処理を示すフローチャート、図12はURLの抽出・解析処理を示すフローチャート、図13は符号化された文字の一覧表、図14は符号化されたURLと復号化されたURLの例を示す説明図、図15は迷惑メールの判定処理を示すフローチャート、図16はデータベースへの登録処理を示すフローチャート、図17は解析結果をユーザに返信する処理を示すフローチャート、図18及び図19は解析結果の一例を示す説明図、図20は解析結果を保存する記憶部の構造を示す説明図、図21はURLに使用可能な文字の具体例を示す説明図である。

【0023】

図1に、一般的な電子メール送受信のシステム構成を示す。図示されているように、電子メール送受信のシステムは、メールサーバ10と、ユーザ端末20と、インターネット30等の通信回線網とを備えて構成されている。インターネット30には、複数のメールサーバ20が接続されており、メールサーバ20間でメールの送受信処理が行われる。

ユーザ端末20は、メールを作成及び送受信する機能を持つメールソフトウェアを備えており、ユーザ端末20がメールサーバ10に接続された状態において、ユーザ端末20はメールサーバ10のメールボックスから、ユーザ宛のメールを取得することができる。なお、ユーザ端末20が携帯電話である場合、メールサーバ10とユーザ端末20とは電通信回線を介して接続される。

【0024】

図2は、本発明の実施の形態における迷惑メール遮断方法及びプログラムが実行されるシステムの構成図である。本システムは、本発明の実施の形態における迷惑メール遮断方法を実行するサーバ装置としてメールサーバ10を備えている。メールサーバ10には、迷惑メール遮断プログラムや、MTA (Mail Transfer Agent) を含むメールサーバソフトウェアがインストールされている。

メールサーバ10は、例えばワークステーションやパーソナルコンピュータ (パソコン) をはじめとするコンピュータ装置により構成される。メールサーバ10は、電子指紋として取得されるメールのハッシュ値や、メールに含まれていたURL情報が格納されるデータベースDB₁～DB₄、ホワイトリスト情報が格納されるデータベースW₁～W₄、メールの解析結果が格納されるデータ格納部11a、11b、メールボックス12、迷惑メールボックス13、ウェブサーバー14、POPサーバまたはIMAPサーバ15を備えて構成されている。また、メールサーバ10は、MTAや各種プログラムに従って各種処理を行う制御部 (図示せず) を備えている。

【0025】

図3に示すように、本例のシステムでは、DB₁～DB₄への登録処理として、標準入力から電子メールを読み込む処理S1と、電子メールの各パートの要素に対する処理S2と、DB参照によるDBへの登録処理S3と、解析結果を登録されたユーザに返信する処理S4と、が行われる。さらに、迷惑メールの判定処理として、標準入力から電子メールを読み込む処理S1と、電子メールの各パートの要素に対する処理S2と、DB参照による判定処理S5と、迷惑メールの遮断処理S6とが行われる。

【0026】

図4に示すように、メールサーバ10は、ユーザ端末20から発信された電子メール、または他のメールサーバから転送されてきた電子メールを受信する。電子メールは、図示しない電子メール保存部に格納されるが、電子メールそのものを受信する前に、受信された電子メールの宛先の判定を行い、その電子メールの宛先がローカルであるかリモートであるかの判定を行う。宛先がローカルである場合には、その電子メールについてローカル処理がなされる。宛先がリモートの場合には、その電子メールは他のサーバへ転送される。

ローカル処理がなされる場合、電子メールは、判定プログラムに入力され、迷惑メールであるか否かが判定される。迷惑メールではないと判定された電子メールは、宛先が指定され、ローカルメーラプログラムによりメールボックス12へ配信される。

【0027】

DB₁～DB₄には、迷惑メールを判定するために必要な各種データが格納されている。

DB₁～DB₄には、電子メールから抽出された情報が格納される。DB₁～DB₄に格納される情報としては、電子メールに含まれるURL文字列(復号化されたもの)、URLから容易に変換できる部分を除いた"host:port/path"の文字列、URLのホストをDNS検索して得られるIPアドレス、メールのパートごとのハッシュ値、の4種類がある。メールのパートごとのハッシュ値は、電子指紋として取得されるものである。

【0028】

DB₁～DB₄に対応して、ホワイトリスト登録用のデータベースW₁～W₄が設けられている。ホワイトリスト登録用のデータベースW₁～W₄には、迷惑メール関連でないことが予め判っている情報が格納される。

W₁～W₄に登録される情報はDB₁～DB₄に格納される情報と対になっており、復号後のURL文字列、"host:port/path"の文字列、IPアドレス、メールのパートごとのハッシュ値、が格納される。ここに記録された各情報に該当したメールについては、迷惑メール判定の対象外とされる。

DB₁～DB₄及びW₁～W₄はハッシュ形式であり、「復号後のURL文字列」、「"host:port/path"の文字列」、「IPアドレス」、「メールのパートごとのハッシュ値」、がキーとされ、「重複登録回数」、「最終更新日時(unix time)」、「最初に登場したメールのアーカイブ名」がデータとして登録される。

【0029】

なお、IPアドレスに関しては、別途、DNSBL用のデータ形式でファイルへ出力される。このようにすると、取得されたIPアドレスをいわゆるDNSBLの仕組みで参照できるようになり、リモートホストでも登録されたIPアドレスをベースにした迷惑メール遮断が可能とされる。最近では、RFC2136を用いて動的にDNSの更新を行うことも可能になり、この技術を利用する場合は、データ形式のファイルが不要となる。

【0030】

本例では、図5に示すように、DB₁～DB₄またはW₁～W₄へのデータの登録・削除の動作時に、ユーザ宛に解析結果レポートを送信することが可能である。メールサーバ10には、URL情報の解析結果が格納されるデータ格納部11a、11bが設けられている。

ユーザ提示される情報は、解析結果をHTMLファイルにすることにより形成される。HTML出力されたファイルは、データ格納部11aに格納される。

また、電子メールそのものがテキストファイルとして出力される。この情報は、データ格納部11bに格納される。データ格納部11a、11bへそれぞれファイルを格納するときは、最初に登場したメールのアーカイブ名がファイル名とされる。電子メールのテキストファイルは、解析結果のページからリンクを張って参照可能とされる。

データ格納部11a及び11bに格納された解析結果は、ウェブサーバ14を介して、外部から閲覧することができる。閲覧を希望するユーザには、これらの情報を閲覧するた

めのURLがメールにより送信される。

ユーザは解析結果を参照し、ホワイトリストに登録すべきものがあるかどうか検討する。ホワイトリストに登録するものがあれば、後述するように、そのURL情報をホワイトリスト登録用のアドレスにメール送信する。

【0031】

メールサーバ10では、DB₁～DB₄、W₁～W₄へのデータの登録及び削除や、データ格納部11a、11bへのデータ出力において、alias（別名指定）を利用したプログラムの呼び出しを行っている。

この場合、特定のメールアドレスにメールを転送することにより、aliasファイルに書かれたプログラムが所定のオプション付きで起動され、転送されたメールがそのプログラムに入力される、という動作がMTAにより行われる。

図6に、本例におけるオプションの一例を示す。また、図7に、メールサーバ10のaliasファイルに書かれているプログラムの一例を示す。

例えば、迷惑メールに含まれるURL情報等を、データベースDB₁～DB₄へ登録する場合は、登録用のメールアドレス（-bオプションの指定されているalias）に対して電子メールを転送することにより行う。

また、特定の情報をデータベースDB₁～DB₄から削除する場合は、その情報をメールに書いて、削除用のメールアドレス（-eオプションの指定されているalias）に送信するだけで良い。

さらに、ホワイトリストデータベースW₁～W₄への登録であれば、ホワイトリスト登録用のメールアドレス（-wオプションの指定されているalias）に対して、本文に登録する情報を書いてメールを送信する。

本例のシステムでは、迷惑メールを判定するための情報として、送信元情報ではなく、誘導先情報であるURL情報と、メールの各パートのハッシュ値を用いているので、結果として、上記のように単純なメール送信によるDBへの登録が可能となっている。

【0032】

DB₁～DB₄は、定期的にメンテナンスが行われ、DBの更新プログラムが定期的に行われる。DBの更新プログラムは、現在日時と、DB₁～DB₄に記録された更新日時を比較し、所定期間以上経過したデータは削除する。例えば、「IPアドレス」は3ヶ月、「メールの各パートのハッシュ値」は3ヶ月、「復号後のURL文字列」は1ヶ月、「host:port/path」の文字列」は2ヶ月で削除される。

【0033】

「メールの各パートのハッシュ値」、「復号後のURL文字列」、「IPアドレス」、「host:port/path」の文字列」は、次のようにして取得される。

「メールの各パートのハッシュ値」は、電子メールがMIME (Multipurpose Internet Mail Extensions) で規定されたパートからなるメールとして送付されたときに、その各パートから得られるものである。

ハッシュ値とは、任意の長さのデータを固定長のデータに投影するハッシュ関数を用いて計算される固定長のデータを指すものである。ハッシュ関数の種類によって、得られるハッシュ値のデータ長は異なる。

本例では、160bitのハッシュ値が得られるSHA1と称するハッシュ関数を用いている。ハッシュ関数としては、上記SHA1の他に、MD5 (128bit)、RIPEMD (160bit)、SHA256 (256bit) などがあるが、本例では、実用上十分な精度があり、計算量も適切なSHA1を用いている。

【0034】

ハッシュ関数を使った場合、データの内容が1バイトでも異なると、全く別のハッシュ値が生成される。このため、ハッシュ値が同一であれば同一のデータであることが判明し、ハッシュ値が異なれば別の内容であると判定することができる。

ハッシュ値を利用することにより、同一のデータが繰り返し送付された場合に、それを検出することができ、迷惑メールとして判定することが可能となる。

【0035】

MIMEメールがMIME-multipartである場合、パートを入れ子にすることが可能であるため、解析は再帰的に行う。

ここで、MIMEメールの構造について説明する。

MIMEメールの最小単位になるパートは、図8に示すように、ヘッダ部、空行、本文から構成されているが、図9及び図10に示すようにマルチパートの構造からなるものがある。この場合、ヘッダ部には、例えば「Content-Type:multipart/mixed」のように記載されている。

マルチパートの構造は、セパレータ文字列を区切りにして、パートの中に、さらにパートを備えた構造となっている。

【0036】

図9及び図10は、入れ子構造のマルチパートとされた例であり、図9の例では、本文の中にさらにパートが一つ設けられている。図10の例では、本文の中にパートが一つ設けられ、このパートの中に、さらにパートが設けられている。セパレータ文字列の前後にマイナス記号を2つずつ付加したものがマルチパート終了を意味する。

また、入れ子構造の他に、パートの中に複数のパートが並列に格納されている構造や、並列に格納された構造と、入れ子構造を組み合わせた構造としたものもある。

迷惑メールでは、上記各パートにURLを含ませて、URLを隠蔽していることがある。このため、各パートをMIME-multipartの構造に沿って復号化し、さらに各パートに含まれるURLを抽出する必要がある。

【0037】

本例のメールサーバ10では、プログラムにしたがって、マルチパートの電子メールの解析が行われる。図11は、マルチパートの電子メールを、その構造に沿って復号化する処理の流れを示すものである。この処理は、図3のS2の処理（パートの要素に対する処理）に該当する。

最初に、本文とヘッダからなるメールが、解析対象のパートとして渡される（ステップS11）。

次に、パートのヘッダ部の解析が行われる（ステップS12）。ヘッダ部には、符号化方式、コンテンツの種類等の情報が記載されている。

ヘッダ部に、例えば「Content-Type:multipart/mixed」と記載されている場合は、電子メールがマルチパートの構造であると判断される。電子メールがマルチパートの構造である場合は、セパレータ文字列を読み取る。

【0038】

ステップS13では、本文のハッシュ計算を行う。

さらに、各パートは、base64やquoted-printableという手法で符号化されていることがあるため、ステップS14で、ヘッダ部に符号化指定が記載されているかどうか判定する（ステップS14）。符号化指定がある場合は（ステップS14；Yes）、ステップS15で復号化処理を行い、本文の復号化を行う。符号化指定があるのに、符号化されていない場合は、復号化処理は行わず、ステップS16に進む。

また、符号化指定がない場合は（ステップS14；No）、ステップS16に進む。

ステップS16では、ステップS12で解析したヘッダ部の情報に基づき、パートがマルチパートの構造であるか否かが判定される。

マルチパートの構造ではない場合（ステップS16；No）、ステップS17に進み、パートが可読なものであるか否かを判定する。パートのメディアタイプが、text/plain、text/html等可読なものの場合には（ステップS17；Yes）、ステップS18でURLの抽出・解析処理を行う。

パートのメディアタイプが可読なものでない場合は（ステップS17；No）、処理を終了する。

【0039】

また、ステップS16で、パートがマルチパートの構造であると判定された場合は（ス

テップS16; Yes)、セパレータ文字列に従って、パートの中に入れ込まれているパートを切り出す処理を行う(ステップS19)。そして、切り出された各パートについて、ステップS11～ステップS19の処理を同様にを行う。

この処理により、電子メールを構成する各パートについて、それぞれハッシュ値が求められるとともに、各パートにURL情報が含まれている場合は、その情報が抽出・解析される。

このように、本例ではMIMEマルチパートが入れ子構造を許していることに対応して、解析手続きを再帰呼び出しして、完全な解析を実現している。

なお、解析結果を収納する記憶部(メモリ)の構造についても、MIMEマルチパートの構造にしたがって入れ子構造になっている。図20(a)は、1つのパートを納める記憶部の構造を示すものである。入れ子構造のパートの場合は、内包するパートを256個まで格納できる。内包されるパートについては、実際には図20(a)、(b)に示すように、この定義で示されている構造体へのポインタとして格納される。

【0040】

次に、図12に基づいて、URLの抽出・解析処理(図11の処理のステップS18)について説明する。

ステップS21では、電子メールに所定のスキームが含まれているかどうかのサーチがなされる。

ステップS21のスキームのサーチにおいて、ターゲットとなるURLは、以下の4種類の仕様(スキーム)に対応するURLである。

「http://」で始まるURL。このURLは、HTTP(Hypertext Transfer Protocol)、すなわちウェブサーバとウェブクライアントの間でHTML文書を送受信するための通信プロトコルに対応しているものである。

「https://」で始まるURL。このURLは、HTTPS(Hypertext Transfer Protocol Security)、すなわち、HTTPとSSL(Secure Sockets Layer)の暗号化機能を組み合わせた通信プロトコルに対応するものである。

「rtsp://」で始まるURL。このURLは、RTSP(Real Time Streaming Protocol)、すなわち、オーディオ・データやビデオ・データを実時間転送するための通信プロトコルに対応するものである。

「ftp://」で始まるURL。このURLは、FTP(File Transfer Protocol)、すなわち、ファイル転送プロトコルに対応するものである。

ここで、URLが1つもない場合には、以下のステップにおけるデータは生成されず、ハッシュ値だけの参照となる。

【0041】

ステップS22では、URLの終点が確定される。

そして、ステップS23では、URLが符号化されているかどうかの判定がなされる。符号化されている場合(ステップS23; Yes)、ステップS24で復号化が行われ、再度、URLの終点が確定される。

ステップS23及びステップS24の処理は、符号化により偽装されたURLに対応するために行われるものであり、この処理により、「復号後のURL文字列」を取得することができる。

符号化の手法として、例えば文字実体参照、数値実体参照、エスケープ符号化がある。

文字実体参照は、DTD(Document Type Definition; 文書型定義)で定義された名前で文字を指定する手法であり、文字コード位置が「&」と「;」で囲まれる記載となる。

文字実体参照の例として、例えば「&」であれば、「&」と表示される。また、例えば「¥」と表示される。また、例えば「<」であれば、「<」と表示される。

数値実体参照において、例えば10進数で指定する場合は、文字コード位置が「&#」

と「;」で囲まれる記載となる。

数値文字参照の例として、例えば「&」であれば「&」と表示される。また、例えば「卒」であれば、「¥」と表示される。また、例えば「<」であれば、「<」と表示される。

【0042】

エスケープ符号化も、本来URLに使用できない文字を取り込むのに使用される枠組みであり、URLの偽装に使用されることがある。エスケープ符号化は、RFC2396に記載されているように、三連文字として符号化されるものである。この三連文字は、「%」文字の後に、2つの16進数字が続く形で構成される。URLにおいてエスケープ符号化がなされた場合は、「w」が「%77」と表示される。

このように、符号化に際しては特定の文字が使用されているため、ステップS23では文字が符号化されているか否かを判定するため、URLのなかに、「&」、「#」、「;」、「%」の文字が使用されているか否かがチェックされる。

【0043】

ステップS24では、符号化された文字を復号するために、文字一覧表が読み込まれる。文字一覧表は、例えば図13に示すように、符号化された文字と、復号した文字とが対になって登録されている。そして、この一覧表を参照して、符号化された文字の復号がなされる。

図14に、復号化の一例を示す。図において、上段が符号化されたURL、下段が復号化されたURLである。例1は数値実体参照による符号化の例、例2は数値実体参照（セミコロンなし）による符号化の例、例3はエスケープ（URI-encode）による符号化の例である。

【0044】

ここで、URIに使用可能な文字で余計な部分を切り捨てる。つまり、抽出されたURLの各部にそれぞれ使用可能な文字のチェックをかけ、使用できない文字が発見されるとそこを末尾と判定する。図21に、末尾判定に使用される文字、ユーザ情報部分に使用可能な文字、ユーザ情報以外の部分に使用可能な文字の具体例を示す。

ステップS25では、URLを要素に分解する処理がなされる。

URLはインターネットで使用される様々なリソースの場所を表すものであり、URLにはリソースを取り出すためのプロトコルやディレクトリ、ポートなどの情報が含まれている。

URLは、一般に次のような形式とされている。

「scheme://userinfo@host:port/path?query」

なお、「userinfo@」、「:port」、「path」、「?query」は省略されていることもある。

ステップS25では、URLを、「scheme」、「userinfo」、「host」、「port」、「path」、「query」に分解する処理がなされる。

【0045】

具体的な例を挙げると、例えば、

「http://ando:password@ns/www.ppml.tv/」

というURLの場合は、次のように分解される。

「userinfo」は「ando:password@ns」。

「host」は「www.ppml.tv」。

「port」は省略されているが、httpなので「80」。ポート番号は、httpなら80、httpsなら443、ftpなら21、rtspなら554、が適用される。

「path」は「/」。

「query」は「なし」。

「scheme」は「http」。

【0046】

迷惑メールに記載されるURLは、特定のURLとして拾われることを防止するため、文字等を追加して異なるURLに見せかけているものがある。

例えば、`userinfo`にわざと「@」を含む文字列を使って、
「`scheme://intrude@intercept@host:port/`」のようにされているURLがある。この場合は、「`intrude@intercept`」が`userinfo`として扱われるべき文字列となる。

また、`userinfo`に「空白文字」を含ませて、
「`http://u s e r i n f o@host:port/`」のようにされているURLがある。この場合は、「`u s e r i n f o`」が`userinfo`として扱われるべき文字列となる。

さらに、`userinfo`に「改行」を含ませて、
「`http://user (改行) info@string@hostname:port/`」のようにされているURLがある。この場合は、「改行」を除き、「`userinfo@string`」が`userinfo`として扱われるべき文字列となる。

【0047】

また、ホスト名部分において偽装されていることがある。

- a. 「`www.ppml.tv`」を、大文字と小文字を入れ替えることにより「`www.PpMl.tv`」としている。
- b. ホスト名である「`www.ppml.tv`」を、IPアドレスである「`210.138.35.27`」としている。
- c. IPアドレス「`210.138.35.27`」を、Hexadecimal (16進数) 形式を用いて「`0xD2.0x8A.0x23.0x1B`」としている。

【0048】

- d. IPアドレス「`210.138.35.27`」を、Octal (8進数) 形式を用いて「`0322.0212.043.033`」としている。
- e. IPアドレス「`210.138.35.27`」を、Hexadecimal 形式 (unsigned long; 符号なし長整数) を用いて「`0xD28A231B`」としている。
- f. IPアドレス「`210.138.35.27`」を、Decimal (10進数) 形式 (unsigned long) を用いて「`3532268315`」としている。

【0049】

上記aのように、「`www.PpMl.tv`」のように大文字で記載されたものについては、「`www.ppml.tv`」のように小文字に変換する。

また、上記b～fのように、IPアドレスを別の形式で書き換えてあるものについて、「`210.138.35.27`」のようにドットで区切られた10進数での表記に統一する。

【0050】

ステップS26では、ステップS25での分解結果から、「`host:port/path`」文字列が生成される。この文字列は、URLの要素のなかで容易に変換できない部分であり、迷惑メールの判定を行う際に有効に用いることができる。

「`http://ando:password@ns@www.ppml.tv/`」の例では、「`host:port/path`」文字列として「`www.ppml.tv:80/`」が生成される。

【0051】

ステップS27では、ホスト名からIPアドレスを取得する処理が行われる。

この処理は、上記ステップS25でIPアドレスが得られた場合には省略される。

ステップS28では、ステップS26でホスト名「`www.ppml.tv`」が得られた場合、このホスト名に基づいてIPアドレスが取得される。IPアドレスは、DNS (Domain Name System) サーバへのアクセスにより得ることができる。

IPアドレスは得られた数だけ全てが取得される。

【0052】

このようにして、URLの抽出及び解析が終了する。ステップS21～ステップS27の処理により、URLが様々な形で偽装されていても、その偽装を解いて、「復号後のURL文字列」、「"host:port/path"の文字列」、「IPアドレス」を得ることができる。

URLの解析が終了すると、解析結果は図20に示す記憶部に保存される。そして、その後、DB₁～DB₄参照による判定処理及びDB₁～DB₄への登録処理等の各処理が行われる。

【0053】

ここで、URLの抽出・解析処理のステップS24の後に行う例外処理について説明する。この処理では、リダイレクターの排除が行われる。

迷惑メールに記載されるURLは、特定のURLとして禁止されることを防止するため、無関係なサイトのリダイレクト機能を用いていることがある。すなわち、本来のURLの前に、別のURLを付加し、この別のURLにアクセスしてきたユーザを、強制的に本来的に見せたいページへ導くものである。

【0054】

リダイレクターを不正に利用したURLは、例えば、
「http://srd.abcde.com/drst/800501378255/
*http://www.365pharm1.com/」
のように記載されている。

上記URLのうち、「http://www.365pharm1.com/」が本来的にユーザに見せたいページを示す部分である。

ステップS41では、URLに記載された「*」の位置が確定される。次いで、ステップS42では、「*」以降に記載されている、本来ユーザに見せたい方のURL（この場合では「http://www.365pharm1.com/」の部分）を抽出する処理を行う。

URLが抽出されたら、URLの抽出・解析処理のステップS26～ステップS28において、URLの解析が行われる。

【0055】

次に、図15において、迷惑メールの判定処理及び迷惑メール送付遮断処理について説明する。判定処理は、図3に示すステップS5及びステップS6に該当する。

判定のフローは、MTAからローカルメイラーの代わりに、以下の処理を行うプログラムが呼び出されて行われる。

ステップS1及びステップS2の処理を経て、「メールの各パートのハッシュ値」、「復号後のURL文字列」、「"host:port/path"文字列」、「IPアドレス」が取得されると、DB₁～DB₄を参照し、一致するデータのスコアを計算し、スコアが1以上であるか否かが判定される（ステップS31）。

スコアが1以上であった場合（ステップS31; Yes）、迷惑メールであると判定され、迷惑メールボックス13にメールが配送される（ステップS32）。

また、スコアが0であった場合（ステップS31; No）、迷惑メールではないと判定され、ユーザのメールボックス12にメールが配送される（ステップS33）。

このとき、メールのヘッダに検出結果（スコア）を付加して配送する。こうすることにより、MUA（Mail User Agent）で、ヘッダ情報を利用した分別等の処理が可能となる。なお、配送先のアカウントを切り替える際は、図6の-u及び-rオプションを使用することにより、切り替えが可能となる。

【0056】

図16は、データベースDB₁～DB₄への登録処理を示すものである。この処理は、図3に示すステップS3に該当する。

登録のフローは、登録用のメールアドレスにメールを転送することで行われる。そうす

ると、登録用のプログラムが呼び出され、登録処理が行われる。

登録処理では、解析結果をもとに、「メールの各パートのハッシュ値」、「復号後のURL文字列」、「"host:port/path"文字列」、「IPアドレス」が、それぞれのデータベースDB₁～DB₄に登録される。

ステップS41では、取得された上記4種類のデータについて、DB₁～DB₄が参照され、当該情報がDB₁～DB₄へ登録されているか否かが判定される。登録があった場合（ステップS41; Yes）は、重複登録回数（カウンタ）を1増加させ、更新日時を更新して再登録する（ステップS42）。

登録がなかった場合（ステップS41; No）は新規登録となる。新規登録されるURL情報は、重複登録回数（カウンタ）が1、更新日時は現在、アーカイブ名は現在処理中のものので登録される（ステップS43）。

【0057】

図17は、解析結果をユーザに返信する処理を示すものである。この処理は、図3におけるステップS4に該当する。

ステップS51では、電子メールがテキストファイルとして出力される。出力されたデータは、「アーカイブ名.txt」というファイル名でデータ格納部11bに格納される。

また、ステップS52では、解析結果がHTML出力される。出力されたデータは、「アーカイブ名.html」というファイル名でデータ格納部11aに格納される。

ステップS53では、解析結果を閲覧希望するユーザに対して、これらの情報を閲覧するためのURLがメールにより送信される。テキストファイルは、解析結果のページからリンクを張って参照可能とされる。

【0058】

図18及び図19は、解析結果の一例を示すものである。

図18に示す例では、送付された電子メールのタイプ（図中の符号A）、本文のハッシュ値（図中の符号B）、各パートのメディアタイプ（図中の符号C）、各パートのハッシュ値（図中の符号D、E、F）、各パートに含まれていたURLの数（図中の符号G、H、I）、送付時のURL文字列（図中の符号J）、復号後のURL文字列（図中の符号K）、URLを分解した結果情報（図中の符号L）、"host:port/path"の文字列（図中の符号M）、IPアドレス（図中の符号N）等の情報が表示されている。

図19に示す例では、送付された電子メールの解析結果に加えて、DB₁～DB₄に登録されるデータについても表示されている。

図19では、送付された電子メールのタイプ（図中の符号A）、本文のハッシュ値（図中の符号B）、メールに含まれていたURL情報の数（図中の符号G）、送付時のURL文字列（図中の符号J）、復号後のURL文字列（図中の符号K）、URLを分解した結果情報（図中の符号L）、"host:port/path"の文字列（図中の符号M）、IPアドレス（図中の符号N）が表示されている。

さらに、抽出されたIPアドレスの重複登録回数（図中の符号O）、更新日時（図中の符号P）、アーカイブ名（図中の符号Q）が表示されている。

【0059】

上記解析結果では、「NEWURL」等、NEWの付いているデータが、新規登録されたことを意味しているので、ユーザはその内容をチェックする。

このとき、DBに登録したくないURL情報（すなわち、迷惑メールとは無関係なURL情報）があれば、そのURL情報はホワイトリストに登録される。

この場合は、迷惑メールと無関係なURL情報をメールの本文に記載し、ホワイトリスト登録用のメールアドレスにメールを送信する。そうすると、図7の「ホワイトリスト登録とレポート」に記載されたプログラムが起動される。そして、該当情報がデータベースDB₁～DB₄から削除されるとともに、ホワイトリストのデータベースW₁～W₄への登録が行われる。

【0060】

ログファイルを作成する際は、ログファイルにはそのメールの含んでいるURL情報とその解析結果及び、詳細なスコア(DBにヒットした数/検出数)が記録される。

ログファイルは、例えば次のような形式で記録される。

X-Picky-Score:101(ip:23/27,hpp:39/43,url:39/43,psig:0/3)

上記の例では、メールサーバで、同じIPアドレス(ip)、または同じhostname:port/path文字列(hpp)、または同じURL(url)、または同じ各パートの電子指紋(psig)のうち、全部で101個がDB₁~DB₄に登録されていたことを示している。

この例では、ip(IPアドレス)については27個検出のうちの23個がDB₃の情報に一致し、hpp(「host:port/path」文字列)については43個検出のうちの39個がDB₂の情報に一致し、url(復号後のURL文字列)については43個検出のうちの39個がDB₁の情報に一致し、psig(メールの各パートのハッシュ値)については3個検出のうちの0個がDB₄の情報に一致したことが示されている。

【産業上の利用可能性】

【0061】

本例のシステムを、有害ウェブサイトへのアクセス制限に用いても良い。

この場合も、上記実施の形態と同様にして、有害ウェブサイトに関するURL情報を取得する。このURL情報を有する端末では、登録されたURL情報に基づいて、有害ウェブサイトへのアクセスがなされようとしている場合に、そのアクセスが制限される。

【0062】

上記構成から把握できる請求項以外の技術的思想を以下に記載する。

(1) 電子メールにより誘導される特定ウェブサイトへのアクセス制限方法であって、

前記電子メールを、メール本文に含まれるセパレータ文字列に従って、メール構造の最小単位であるパートに分解するメール解析工程と、

前記分解したパート毎に電子指紋を取得する電子指紋取得工程と、

前記パートに含まれるURL情報を取得するURL情報取得工程と、

過去の電子メールから取得・蓄積された電子指紋またはURL情報が格納されたデータベースを参照するデータベース参照工程と、

前記電子指紋取得工程で取得された電子指紋または前記URL情報取得工程で取得されたURL情報が、前記データベースに迷惑メールに関連する電子指紋またはURL情報として格納されていた場合に、前記URL情報を有害ウェブサイトに関する情報と判定して、該有害ウェブサイトへのアクセスを制限するアクセス制限工程と、を備えたことを特徴とする特定ウェブサイトへのアクセス制限方法。

【実施例】

【0063】

迷惑メールに関するデータを提供するサイト(例えばスパムアーカイブ; <http://www.spamarchive.org/>)を利用して、本システムの機能を評価した。

迷惑メールを1通ずつ解析し、迷惑メールとして登録するという手順で、34日分、37,000通あまりをDBに登録した。

その結果、URLからIPアドレスが取得できたものに限ると、31日目のデータで99.58%の迷惑メールを遮断することができた。これは一般的な迷惑メール遮断の手法を上回る検出率である。

【0064】

また、37,000通のメールを解析した結果、各DBに登録されているデータエントリーは、

URL文字列のDB:68,473件

host:port/path文字列のDB:40,526件

メールの各パートのハッシュ値のDB:37,424件

IPアドレスのDB:5,202件

となった。

このなかで、検出に最も貢献したのは、IPアドレスのDBであった。

このように、本例のシステムでは、小さなサイズのDBで高い検出効率を達成することができるため、大規模サイトへの適用も可能である。

また、従来の仕組みのように、発信元のホストやIPアドレス、発信者のドメインを利用して迷惑メールを判定する手法では、迷惑メールの発信元として登録されてしまうと、それ以降その発信元からはメールが届かなくなってしまうが、本例のシステムでは、誘導先のURL情報さえ消せば確実にメールは到達するので、実用上、より弊害の少ない安全なシステムになっていると言える。

【図面の簡単な説明】

【0065】

【図1】一般的な電子メール送受信のシステム構成を示す説明図である。

【図2】本発明の実施の形態における迷惑メール遮断方法及びプログラムが実行されるシステムの構成図である。

【図3】迷惑メール検出の手順を示すブロック図である。

【図4】メール受信からメールボックス格納までの流れを示す説明図である。

【図5】メール受信から解析結果提示までの流れを示す説明図である。

【図6】プログラムのオプションの例を示す一覧である。

【図7】実際のaliasの例を示す一覧である。

【図8】MIMEメールの最小単位になるパートを示す説明図である。

【図9】マルチパートの構造からなるMIMEメールを示す説明図である。

【図10】マルチパートの構造からなるMIMEメールを示す説明図である。

【図11】マルチパートの電子メールの復号化処理を示すフローチャートである。

【図12】URLの抽出・解析処理を示すフローチャートである。

【図13】符号化された文字の一覧表である。

【図14】符号化されたURLと復号化されたURLの例を示す説明図である。

【図15】迷惑メールの判定処理を示すフローチャートである。

【図16】データベースへの登録処理を示すフローチャートである。

【図17】解析結果をユーザに返信する処理を示すフローチャートである。

【図18】解析結果の一例を示す説明図である。

【図19】解析結果の一例を示す説明図である。

【図20】解析結果を保存する記憶部の構造を示す説明図である。

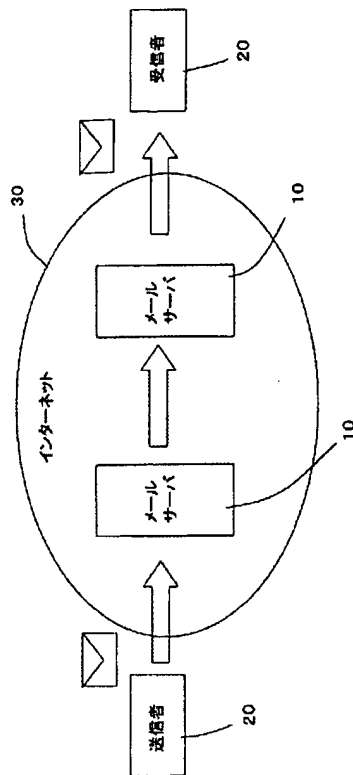
【図21】URLに使用可能な文字の具体例を示す説明図である。

【符号の説明】

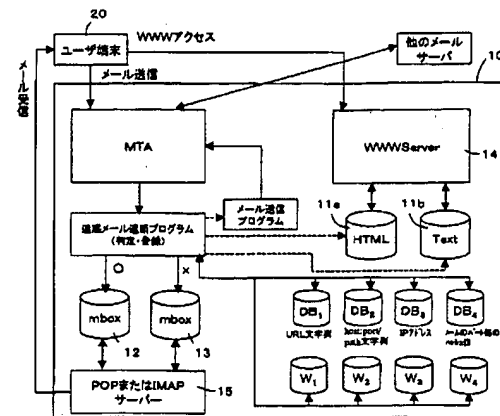
【0066】

10 メールサーバ、11 メール受信部、12制御部、13 記憶部、14 メールボックス、15 迷惑メールボックス、16 リモート処理部、20 ユーザ端末、30 インターネット

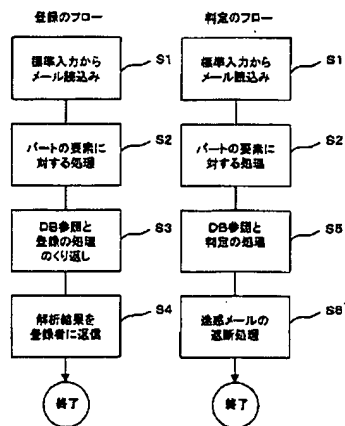
【図1】



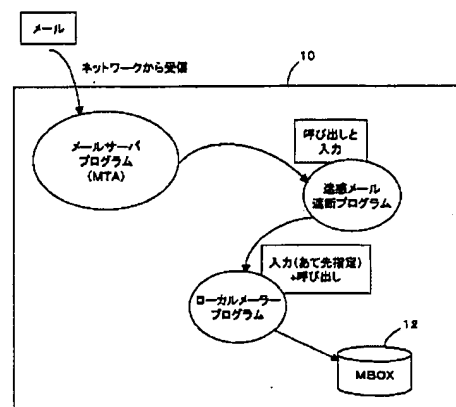
【図2】



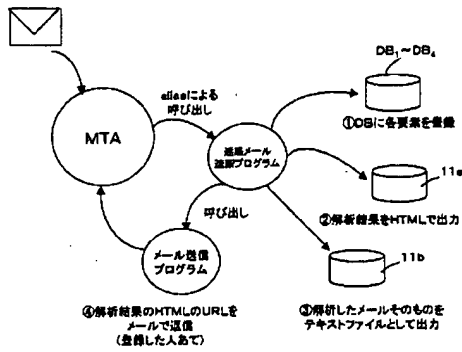
【図3】



【図4】



【図5】



【図6】

```

-a: write results to html and report
-b: update DBs with analysis result
-c: check DBs (use this with -t or -u -r)
-d<n>  :set debug level
-e: erase DB entries and write to html
-w: update white-list DBs
-t<user>: all articles to user by mail, local
-u<user>: passed article to user by mail, local
-r<user>: rejected article to user by mail, local
-l<logfile>: set logfile
-v: verbose command line output of analyze result
-q: quiet mode

```

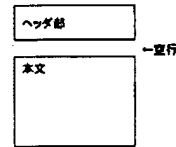
【図7】

```

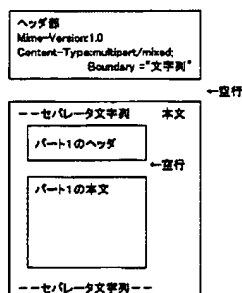
#実行モード
picky-test: "/usr/local/ppml/picky -a -q -t picky -i /var/log/picky. log"
#登録とレポート
picky-entry: "/usr/local/ppml/picky -d 0 -a -b -q -i /var/log/picky. log"
#削除とレポート
picky-check: "/usr/local/ppml/picky -d 0 -a -q -i /var/log/picky. log"
picky-erase: "/usr/local/ppml/picky -d 0 -a -q -i /var/log/picky. log"
#ホワイトリスト登録とレポート
picky-write: "/usr/local/ppml/picky -d 0 -a -w -q -i /var/log/picky. log"

```

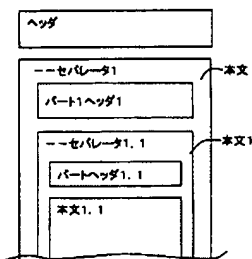
【図8】



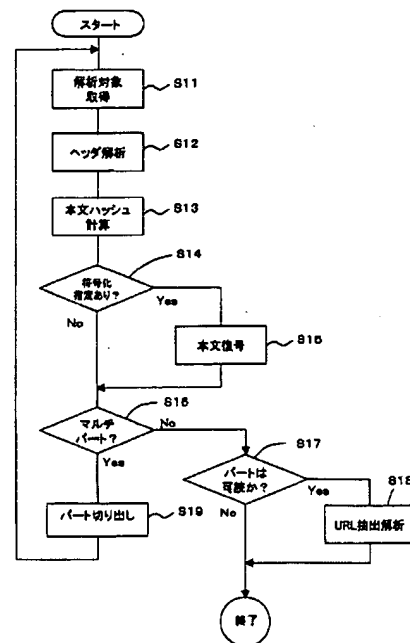
【図9】



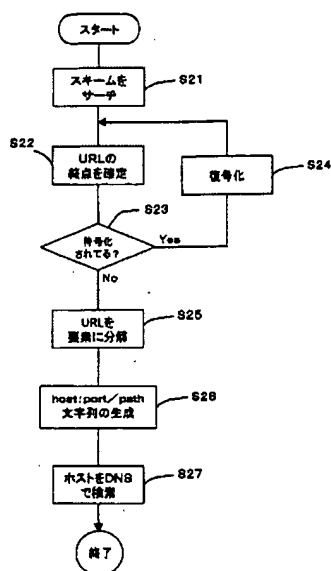
【図10】



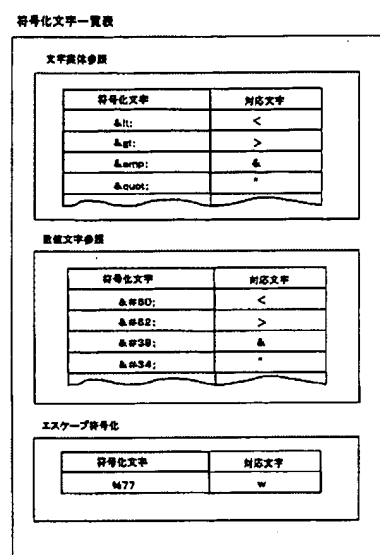
【図11】



【图12】



【図13】



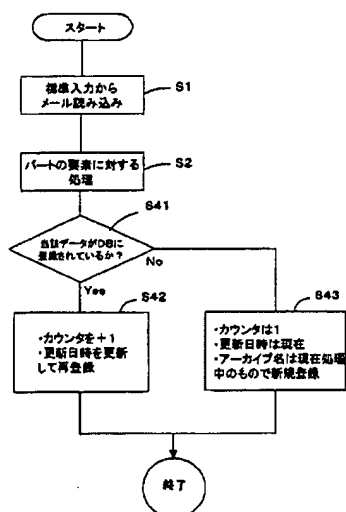
【図14】

(例1)
http://8666881178211218104881038728478881111887888788111188777/a.html
http://BuyhgN0w.CoM6/a.html

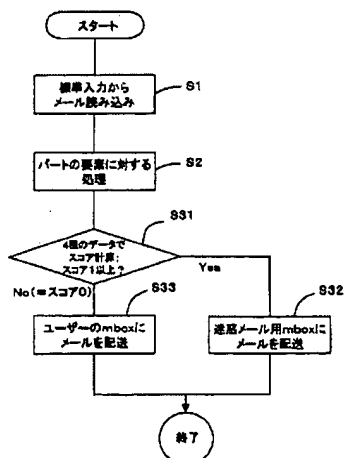
(例2)
http://888888117881181044891038872878881111887888788111188777/b.html
http://BuyhgN0w.CoM6/b.html

(例3)
http://treasure.sink06666666444233472E666666/cd/ads.jpg
http://treasure.sink06666666447.info/cd/ads.jpg

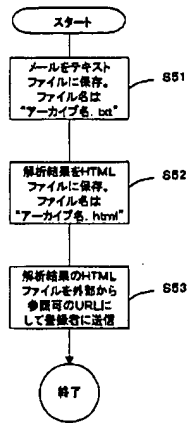
【图16】



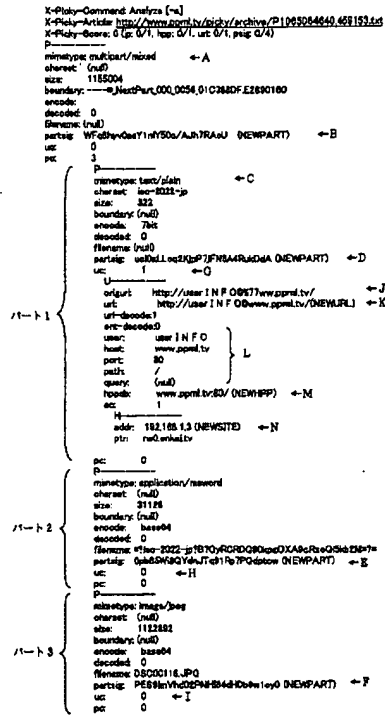
【図15】



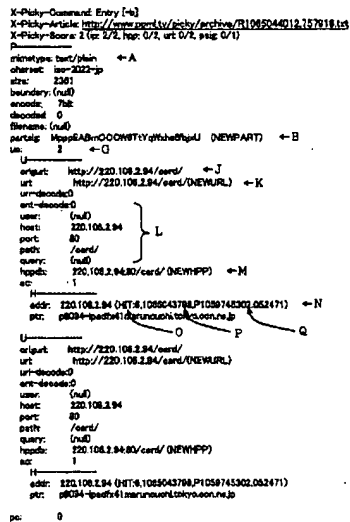
【図17】



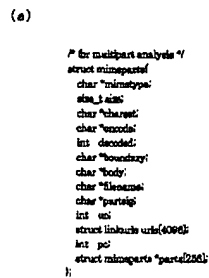
【図18】



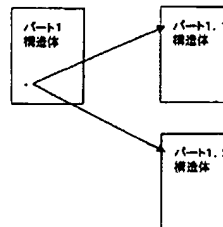
【図19】



【図20】



(a)



【図21】

最初の文字列
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ123456789-~!@#\$%&*"
宛先情報の文字列化後の文字列
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ123456789-~!@#\$%&*"
エスケープ文字列化後の文字列
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ123456789-~!@#\$%&*"

ユーザ情報部分に使用可能な文字
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ123456789-~!@#\$%&*"
ユーザ情報以外の部分に使用可能な文字
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ123456789-~!@#\$%&*"